Tu Luan

Senior Thesis

Professor Steven Lindell

Feb 07, 2018

Abstract

A little lost on the definition of formal language. Not sure what you mean her by “used to specific to it”. What is it referring to?

Formal language is referred to a set of strings of symbols and rules that used to specific to it. Formal language theory mainly studies the internal and syntactical structure pattern of formal language. In computer science particularly, formal languages are applied on defining the grammar of programming languages and formalized natural languages in which concepts that are related to meanings or semantics, are represented by the words of those languages. To be more specific, the source code parsing of compilers of many modern languages involve formal language theory, where the formal language theory ensures the validity of an incoming piece of source code. Additionally, in theoretical computer science, decision problems in computational complexity are always defined in formal language. Besides the field of computer science, formal language theory is also widely used in linguistic and math: Formal language theory is applied in mathematical logic where a formal system contains a formal language and a deductive apparatus.

nice(concise and clear) examples for how formal language theory can be applied

The main subject of this paper is the regular expression, or regex, a subcategory of formal language. In formal language theory, regular expression is a sequence of characters with searching pattern. The concept of regular expression was brought up by American mathematician Stephen Cole Kleene in 1950s. He described regular expression with mathematical notations and

regular expression was called “regular rules”. This theory was soon applied to theoretical computer science, in the automata theory. Regular expression became popular in real word application in1960s and it was used to math patterns of text, for example in text editors.

is this a typo? You want to say “match” instead of “math”?

Our major goal of the research is to develop a systematic definition of regular expression in the form of monadic second order logic. And we want to prove our definition of regular expression has the equivalence relationship to the traditional definition we have in computer science. The major concepts involved this research come from monadic second order logic from philosophy, regular expression from computer science, and set theory from mathematics. The goal of this research is inspired by the correlation of regular expression and monadic second order logic. When Professor Lindell and I investigated the concept of regular expression, we noticed a concept overlapping between regular expression and monadic second order logic and therefore we are interested in whether we can express regular expression in a proper definition of monadic second order logic. Before digging into monadic second logic, we previously tried to express regular expression in the form of mathematics algebraic form. However, due to a lack of proper proofs, we decided to move to the current direction of monadic second order logic.

Maybe explain what is the traditional definition we have in computer science?

involved in?

Maybe give a short definition/explanation on monadic second order logic, as it might not be familiar to a general cs major

interesting and clear motivation

do you mean mathematical algebraic form? Maybe give a definition could help clarify

Is your thesis potentially involve some results(maybe as significant failure) about your attempts using mathematics algebraic form?

In order to complete this work, we plan to investiage more concepts and knowledge of monadic second order logic and form a more complete understanding of our subject. This may involve readings and practicing on the materials given my advisor, Professor Steven Lindell.

Provide more details as how you plan to carry out the research. As what kind of readings you might be focus on, what consists as practice? Are you writing any code/implementation? Are you writing proofs? etc…